



**University of
Zurich^{UZH}**

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2004

Requirements for mobile learning games shown on a mobile game prototype

Göth, Christoph ; Häss, Urs-Peter ; Schwabe, Gerhard

Abstract: Mobile technologies offer the opportunity to embed learning in a natural environment. This paper describes the design of the MobileGame prototype. MobileGame explores the opportunities to support learning through an orientation game in a University setting. The paper first introduces the scenario and then describes the general architecture of the prototype. The second half of the paper focusses on requirements that have evolved during the design, implementation and testing of the prototype: Supporting work on the move poses difficult interface questions, the accuracy of current outdoor and indoor positioning systems is still problematic and the game requires near real-time response time. This requires a distribution of functionality and data between the server and the client and a careful design of the interface. The success of the game heavily depends on a motivating design of the game itself (i.e. the setup and its rules). A surprising number of user roles surface once the game is implemented in a natural environment.

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-61359>

Conference or Workshop Item

Accepted Version

Originally published at:

Göth, Christoph; Häss, Urs-Peter; Schwabe, Gerhard (2004). Requirements for mobile learning games shown on a mobile game prototype. In: Proceedings MLearn 2004 conference, Rome, 1 January 2004, 95-100.

Requirements for mobile learning games shown on a mobile game prototype

Christoph Göth

Information Management
Research Group
Department of Informatics
University of Zurich
goeth@ifi.unizh.ch

Urs-Peter Häss

Information Management
Research Group
Department of Informatics
University of Zurich
haess@ifi.unizh.ch

Gerhard Schwabe

Information Management
Research Group
Department of Informatics
University of Zurich
schwabe@ifi.unizh.ch

Abstract

Mobile technologies offer the opportunity to embed learning in a natural environment. This paper describes the design of the MobileGame prototype. MobileGame explores the opportunities to support learning through an orientation game in a University setting. The paper first introduces the scenario and then describes the general architecture of the prototype. The second half of the paper focusses on requirements that have evolved during the design, implementation and testing of the prototype: Supporting work on the move poses difficult interface questions, the accuracy of current outdoor and indoor positioning systems is still problematic and the game requires near real-time response time. This requires a distribution of functionality and data between the server and the client and a careful design of the interface. The success of the game heavily depends on a motivating design of the game itself (i.e. the setup and its rules). A surprising number of user roles surface once the game is implemented in a natural environment.

Keywords: Mobile & Wireless Games, Mobile Learning, E-Learning,, Computer Supported Cooperative Play, Positioning Systems

1. Introduction

Today, we work increasingly with mobile techniques. There are mobile phones to communicate everywhere, laptops to work wherever we want, and Personal Digital Assistants to get access to information whenever we want. It's just a question of time when mobile techniques will be used for other aspects of our life. One of these is learning. Sharples ([SCW02]) mentions that the technique for mobile learning is now available. Research in the area of mobile learning, however, is still at the beginning, but the outcome shows that mobile technique provides a new way of learning.

One of these new ways could be the integration of learning and playing supported by a mobile environment. There are a few systems that integrate playing and learning such as the Cooties Game or Geney ([SoKe03]) but they focus on role-play or simulation. Prototypes and commercial products of location-based games in a real life environment, like CYSMN ([BAF+03]), Pirates ([BFHL01]) or Mogi ([Hal01]) show that people like to play with the new options but these games are only focused on entertainment. Below we describe a scenario, which shows a new way of learning combining location-based games and mobile learning. This scenario was implemented in a prototype for testing. We describe the architecture of this prototype before we show requirements which we found in our first test with users.

2. The Scenario

Mark Finder is a new student at the University of Zurich and has just arrived for the orientation day. This year, the traditional orientation rally is electronically supplemented with handheld devices. The orientation rally is a fun event intended to get to know the university and surroundings. Therefore, the rally will lead all participants through a parcours with several tasks to fulfil at certain spots. He is asked to fill in an online-form with his personal profile (nationality, gender, age, personal interests, hobbies, favourite food etc.). All new

Participating students are automatically grouped by their profiles in order to obtain homogeneous groups. Mark meets five other students with similar interests in theatre, jazz, history and biking. Each group receives a handheld computer.

During the orientation rally, each group receives different tasks referring to significant places, people and events (explained below). The handheld device shows the current position of the group on the digital map of the university (see Figure 1). When the group enters a building the outdoor map switches to an indoor map of the building the group just entered. The whole rally is structured as a cooperative and competitive game. Competition is based on hunting rules: Each group tries to catch another group and, equally, is hunted by another group¹. The handheld device shows each group where its hunter and its prey are located. Cooperation forces group members to meet members from other groups as well as teachers and to exchange information with them - again they are supported with location based information on their displays. The tasks given to them provide them with basic information on University live. There are the following types of tasks:

- *Significant place tasks:* The students have to find important places such as the library, the cafeteria or the laboratories. At each location they have to perform a typical task (find a book, have lunch, etc.). The specific tasks are context-dependent (they depend not only on the location, but also on the time of the day or they build on the activity of some previous group). The task execution is supported by the handheld device (e.g. serving as a front end to the library information system or providing them with needed information).
- *Significant people tasks:* The students have to find important people of the university and have to interview them on their tasks (the president, the study coordinator, the caretaker...- These people either participate in the game or are played by elder students). If those people are typically mobile, they can be located by a mobile device.
- *Significant event tasks:* The significant events can be scheduled or by surprise. Scheduled events include introductory lectures and courses. Here tasks relate to the organization of studies (e.g. set up a course schedule or how to find important information) and some initial content. Unscheduled events include "spontaneous" welcome parties by student groups, but also the sign up of each group member to important university services (e.g. computer account, library card).

Each task requires the group to answer one or two simple questions displayed on the handheld device. For example, one task might be to find the cafeteria. There they get the question "What is the price of an apple?". They won't get the next task until the correct answer is given.

In addition to the game, the students can annotate real objects with virtual “post-it’s”. Other groups can read these short messages and answer with their own post-it’s. Spectators are also integrated. They can watch the game in a web browser. The groups and the spectators can use the integrated instant messenger to communicate with each other.

The mobileGame scenario substitutes the teacher by a PDA, giving tasks to the student. This corresponds to the demand of the constructivism theory for the teacher to act as a coach. The learner is encouraged by the tasks to act on his own, in the real world context of the university. He individually constructs his own knowledge at the university. Additionally, the game is played by students with the same interests. This links the game with a suitable social setting, as postulated by the situated cognition theory.

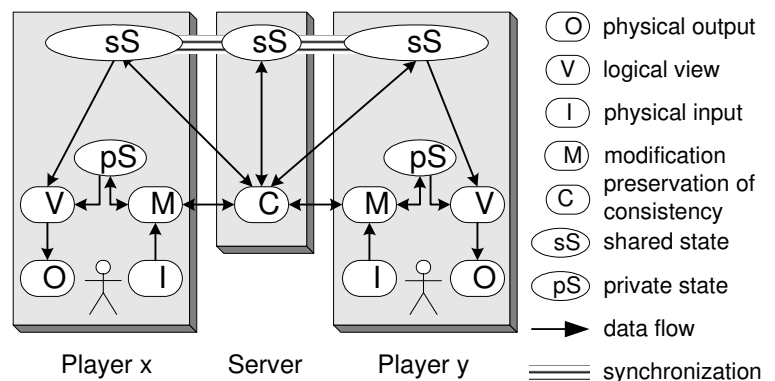


Figure 1: Playing the game

¹ The didactic reason for hunting rules is to keep the groups moving. Of course there need to be hunting free areas and times, e.g. during courses.

3. The mobileGame prototype

The scenario describes a sort of synchronous collaboration between the playing people. So we decided to derive the architecture of the mobile game prototype from a common cscw architecture concept for synchronous collaboration (see Figure 2) ([ScSc01]). There are the clients with their own private state of the ongoing game. Mostly all relevant data is stored on the client side so the player could act even if there is no connection to the server available. The server works as the central coordination point for the whole game. Dynamic changes in the game are transferred to the server which distributes the data. Awareness over the accuracy of the transferred information is handled by a timestamp mechanism. Thus, a player knows always if he works on "fresh" data or if his messages from other players are obsolete.



3.1. The technique of the prototype

The architecture integrates three components; the mobile client on the PDA, the web client for a browser and the server. The mobile client runs in a Java virtual machine² on the PDA. It communicates with the server over RMI. For example, the client receives information about the current position of the group, new tasks to solve, events or messages. The client displays the data on the screen of the PDA. If the group catches other groups or solves tasks, this information is sent back to the server. Annotations are displayed in the Pocket Internet Explorer. The browser exchanges data with the server over HTTP. On the PDA, there also runs an instance of the Ekahau client. This program is independent from our own code and communicates with the Ekahau positioning engine autonomously. If the playing ground is not completely covered by Wireless-LAN, the client can also get positioning information over a serial connection to a GPS receiver.

The web client is a Java applet running in the Java virtual machine of a web browser. It requires the Java 1.4 plug-in from Sun. The applet shows the current status of the game and allows chatting with the playing groups. The communication with the server is realized over HTTP.

The server components are also written in Java. As described above, it communicates with the different clients over RMI and HTTP. For the communication over HTTP, we use the Jetty servlet engine, a simple but efficient servlet container. MySQL and Apache OJB³ are used for the persistent storage of game data. The server gets the positioning information of the playing groups from the Ekahau positioning engine which itself gets them from the Ekahau clients running on the PDAs.

3.2. Positioning of the client

There are two ways to localise the position of the client. The first one is used in outdoor areas, where no WLAN is available. A GPS receiver is plugged to the PDA which computes his position with the GPS signals stand-alone. The problem of GPS positioning is its inaccuracy. The accuracy ranges from three to hundred meters, depending on the weather and the surroundings. There are many unwanted side effects like jumping positions over the whole digital map or groups were not available at the point shown on the digital map.

² This JVM has to support the Personal Java 1.2 specification (<http://java.sun.com/products/personaljava/>)

³ "ObjectRelationalBridge (OBJ) is an Object/Relational mapping tool that allows transparent persistence for Java Objects against relational databases." See <http://db.apache.org/ojb/>

The second, much more accurate way, is to use the WLAN to compute the position. We used the Ekahau Positioning Engine for this purpose. The Ekahau client on the PDA sends the information of the available WLAN access points to the Ekahau Positioning Server. The server computes the position, which is read from the mobile-Game Server. The accuracy is from one to three meters, if four access points are available. The disadvantage is that the engine needs to be calibrated before you can use it. This calibration took up to one hour per floor of a building, so you can't use it out of the box.

4. Requirements and lessons learned

The following requirements are based on a user test, which took place at the University of Koblenz. The test was integrated in a presentation day of the faculty of computer sciences. The mobileGame was shown to the visitors and they could play a little game, in which they had to find a hidden PDA.

4.1. Work on the move

The players get a PDA showing the map of two floors, their own position and the position of the PDA, which they have to find. Although the map size was 50 % of the PDA screen and the floor was only a straight corridor the players mentioned that it was not possible to move and navigate with the PDA at the same time. If they wanted to look for their current position they had to stop and look at the PDA. Even though they don't have to interact with the PDA they have to look down and compare the map on the screen with their surrounding. It's really hard for people to work on the move.

In the mobileGame, the players have to position themselves constantly. Of course, they don't want to stop every 10 seconds and look at the PDA. Therefore, we have done some experiments with head mounted displays presenting information close to the line of sight. This option, however, has two disadvantages: the players have to wear big glasses and not everybody likes to look like a cyborg or a robot. The other disadvantage is the distorted field of vision. There are two kinds of head mounted displays. The first is a little display, which you plug on your normal glasses. The second are big glasses with integrated video in- and output. Two video cameras in the glass record parts of the field of view and display it on two screens in front of your eyes. This can be augmented with additional information. Depending on the model, you have to refocus your eyes on the little display on your glasses or you wear big glasses which only present a restricted field of view.

We believe that an audio interface, which gives the user the needed information through headphones, may be the answer to that problem. Björk et al ([BFHL01]) have experimented with audio for status information in their game Pirates!. The result was that often this information could not be heard because the noise level within the gaming area was too high. So there must be a trade-off between the transmission of information through transient audio and persistent visualisation on the PDA screen.

4.2. Accuracy of positioning

One of the biggest mobile games was "Can you see me now?" designed by Benford et al ([BAF⁺03]). Mobile actors hunt virtual players in Sheffield. Their positioning system for the actors is based on GPS. Their main problem was accuracy of up to 106 meters, but with some tricks the actors have managed this problem. Our prototype uses GPS and WLAN for positioning, as described above, with the result of an accuracy of three to five meters. The players, who need the positioning information for their orientation, have told us that the accuracy was quite good as long they only need to know their approximate position. As soon as they have to know the exact position of an object, in our test the position of the hidden PDA, the accuracy was not sufficient. They had to search in up to three rooms for the PDA.

For each domain of positioning you need a separate system. Global and local systems can be integrated to get a much better result for the players. We are planning to add a third system beside the GPS and the WLAN system, which is based on RFIDs. Points of special interests can be tagged with RFIDs so a player, who reaches this point, receives a signal and knows his position is accurate within up to one meter.

4.3. Offline areas and short reaction time

One of the main reactions of the players in the test was that the update time of their position is much too slow. We update their position every three seconds but in this time the players move five or more meters. This describes one of the most important requirements of mobile games. Mostly you don't have a area-wide WLAN, so the mobile clients are not always connected to the server. But the players want a near real-time reaction of the client. Game objects, which are changing all the time, like the position of the gamers, have to be updated by client as fast as possible. This means on the one hand, that the mobile game must have a very good caching algorithm, and on the other hand, that a very efficient data transmission strategy is needed.

In our prototype all static information, such as the game maps, are stored on the PDA. Only the dynamic information such as a new position or a new task, are transmitted. The PDA gets all information from the server that he need for offline work, like the answer of the current task, so that the players can interact even if they are in an area without WLAN. Additionally, they get the status information about their network status and the status of other players, they want to interact with.

4.4. Easy to use interface

Figure 3 shows the new and the old interface. The first interface uses drop down menus like desktop programs. The observation of the players shows that navigation with the drop down menu and the pen of the PDA is not really intuitive. The use of the PDA is much more like a use of an automat. So we redesigned the GUI and substituted the old menu with a button bar. Now all functions can be reached by one or two clicks. We also use symbols instead of text because they are smaller than text and the users understand symbols much better.

One big problem is still the interface for answering the questions of the task. On the one hand, you can use open questions, which gives the game designer a broad range. Then the users have to write the answer down with the pen and a little virtual keyboard on it, which is hard to use. On the other hand, you can use multiple choice questions, which restrict the game design, but users can answer questions by simply clicking. At this point in time we support open questions, multiple choice questions and questions with sliders, where you can ask for numbers. We think an audio interface could help here too.

4.5. Roles in the game

In the test in Koblenz, we used only one client, the mobile Client. The players as well as the game master use the same functions with the same client. Some functions were spontaneously "re-defined" because there were no adequate functions given by the client. After the test, the results were discussed in a meeting with experts. A result of this discussion is that a mobile game needs to support many roles in the game. There are the player, the game master, the spectators, the visitors, the tutors and the actors for special events. Also there are the game designer and administrator beside the game. All these persons need a special client with special functions. At this point in time we only have a client for the players and for the spectators.

5. Ongoing Research

The next step of our research is a big user test with new students at the University of Koblenz at the end of April. As described in the scenario, new students should learn how to live on campus and where to find the most important places. In the presentation at the MLEARN, a first result of this test can be presented. After the analysis of the test, the prototype should be revised. In autumn, the revised version is planned to be tested at the University of Zurich.

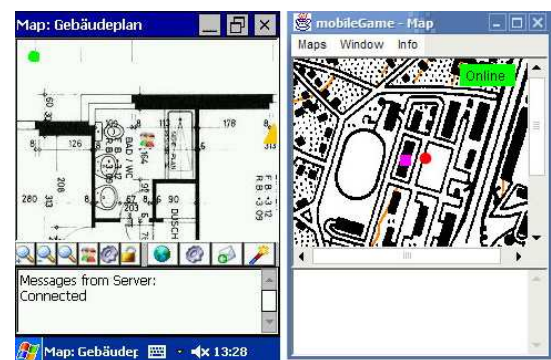


Figure 3: The new and the old GUI

6. References

- [BAF⁺03] Steve Benford, Rob Anastasi, Myartin Flintham, Adam Drozd, Andy Crabtree, Chris Greenhalgh, Nick Tandavani, Matt Adams and Ju Row-Farr (2003) "Coping with Uncertainty in a Location-Based Game" IEEE Pervasive Computing journal, July-September 2003
- [BFHL01] Steffan Björk, Jennica Fals, Rebecca Hansson and Peter Ljungstrand (2001): "Pirates! Using the physical World as a Game Board" Paper at Interact 2001, IFIP TC.13 Conference on Human-Computer Interaction, July 9-13, Tokyo, Japan.
- [Hal01] Justin Hall (2001): "Mogi: Second Generation Location-Based Gaming" <http://www.thefeature.com/article?articleid=100501&threshold=-1>
- [SaKe03] Savill-Smith, C. & Kent, P. (2003): „The use of palmtop computers for learning. A review of the literature.” London: LSDA.
- [ScSc01] Jan Schümmer and Christian Schluckmann (2001): „Synchrone Softwarearchitekturen“ In Schwabe, Streiz, Unland (2001): „CSCW-Kompendium“ Springer
- [SCW02] Mike Sharples, Dan Corlett and Oliver Westmancott (2002). "The design and implementation of a mobile learning resource". Personal and Ubiquitous Computing, 6:220–234.